UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/910,519	07/20/2001	Wylci Fables	SSR-P1D	3023

			EXAMINER
26793	7590	05/12/2005	RAMPURIA, SATISH

LEIGHTON K. CHONG
GODBEY GRIFFITHS REISS & CHONG
1001 BISHOP STREET, PAUAHI TOWER SUITE 2300
HONOLULU, HI 96813

ART UNIT	PAPER NUMBER
2191	

DATE MAILED: 05/12/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

		Application No.	Applicant(s)
Office Action Summary		09/910,519	FABLES ET AL.
		Examiner	Art Unit
		Satish S. Rampuria	2124

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on <u>31 January 2005</u>.

2a) ☐ This action is **FINAL**.   2b) ☒ This action is non-final.

3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) <u>21-40</u> is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) ☐ Claim(s) _____ is/are allowed.

6) ☒ Claim(s) <u>21-40</u> is/are rejected.

7) ☐ Claim(s) _____ is/are objected to.

8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☒ The specification is objected to by the Examiner.

10) ☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
    Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

Response to Amendment

1. This action is in response to the amendment received on Jan 31, 2005.

2. Claims 21 and 35 are amended.

3. Claims 21-40 are pending.

Specification

4. Applicant is required to update the status (pending, allowed, etc.) of all parent priority

   applications in the first line of the specification. The status of all citations of US filed

   applications in the specification should also be updated where appropriate.

Double Patenting

5. The nonstatutory double patenting rejection is based on a judicially created doctrine
grounded in public policy (a policy reflected in the statute) so as to prevent the
unjustified or improper timewise extension of the "right to exclude" granted by a patent
and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d
1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed.
Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*,
422 F.2d 438, 164 USPQ 619 (CCPA 1970); and, *In re Thorington*, 418 F.2d 528, 163
USPQ 644 (CCPA 1969).
A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to
overcome an actual or provisional rejection based on a nonstatutory double patenting ground
provided the conflicting application or patent is shown to be commonly owned with this
application. See 37 CFR 1.130(b).
Effective January 1, 1994, a registered attorney or agent of record may sign a terminal
disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37
CFR 3.73(b).

Claims 21, 22, 24, 25, 26 and 35 rejected under the judicially created doctrine of

obviousness-type double patenting as being unpatentable over claims 1-4, of U. S. Patent No.

6,282,697 (hereinafter called '697). Although the conflicting claims are not identical, they are

not patentably distinct from each other because of the following observation.

Instant Claim	*'697 Claim*
21. A distributed computing method comprising the steps of: **creating a plurality of software entities ("molecules") each of which is** to be deployed in a remote computing environment and is **configured with signal handling software micro-components**	1. A computer processing method comprising the steps of: **creating a plurality of software entities ("molecules") each of which is configured with software micro-components including a signal handler**, at least one input handler, at least one output handler, an interface handler, and at least one method handler for an associated method, said input handler, output handler, and signal handler being operative
for sending and receiving communication signals to or from a source externally of the respective molecule, said signal handling micro-components of each molecule being operatively connected a selected one of a plurality of method handling software micro-components for processing input data by a respective selected method in the given remote	**for sending and receiving communication signals to or from another molecule or logic web externally of the respective molecule and being operatively connected to the other micro-components**, said at least one input handler being operative for queuing input data, said interface handler being operative for determining when a predefined input

computing environment in which said molecule is deployed and providing	conditional for required input data to be received by said input handler is fulfilled and then invoking said method handler, said method handler being operative for invoking said associated method for processing the input data, and said at least one output handler being operative for outputting **a result of the**
a resulting output of processing the input data;	**processing of input data by said method;**
deploying a first molecule with a method handling software component for a specific selected first data processing method on a respective one of a plurality of remote computing environments, together with a library containing a plurality of method handling software components for a corresponding plurality of data processing methods which may be selectively invoked in the remote computing environment;	storing the created molecules in a library for later run time use;
	deploying on a given computing resource a logic web comprising a plurality of molecules selected to perform a given processing task by invoking a first molecule to be retrieved from the library and executed on the given computing resource; and
wherein each first molecule deployed its respective computing environment invokes a	said first molecule invoking one or more other molecules to incrementally extend said logic web "on the fly" by said signal handler of said first molecule sending a communication

next subsequent molecule with a method handling software component retrieved from the library for implementing a next selected in response to receipt of a communication signal from the external source, thereby creating a "logic web" of molecules of successive data processing method in successive layers of incremental processing steps to incrementally extend said logic web "on the fly" in response, at least in part, to communication signals from the external source.	signal to invoke said other molecule(s).
22. A distributed computing method according to Claim 21, **wherein each logic web in each computing environment performs its data processing functions in its respective computing environment autonomously,** and returns an output which is desired to be obtained from that computing environment.	2. A computer processing method according to claim 1, **wherein said creating step includes creating a plurality of logic webs each having its web of software entities configured to perform a data processing task with a computing resource autonomously,** and said deploying step includes deploying the plurality of logic webs with respective ones of a plurality of computing resources, respectively.

24. A distributed computing method according to Claim 23, **wherein the computing environments are a plurality of computing sites distributed on a network, and the logic webs return their outputs by sending sirals on the network.**	3. A computer processing method according to claim 2, **wherein said plurality of computing resources are distributed on a network in a distributed computing environment.**
25. A distributed computing method according to Claim 23, **wherein the computing environments are a plmality of computing resotlrces in an array of processing units (CPUs) operated in parallel in a parallel processing environment.**	4. A computer processing method according to claim 2, **wherein said plurality of computing resources include an array of central processing units (CPUs) in parallel in a parallel processing environment.**
26. A distributed computing method accordinq to Claim 21. wherein said **software micro-components include a signal handler, at least one input handler, at least one output handler, an interface handler, and at least one method handler for an**	1. A computer processing method comprising the steps of: creating a plurality of software entities ("molecules") each of which is configured with **software micro-components including a signal handler, at least one input handler, at least one output handler, an interface handler, and at least one method handler**

mssociated method,	for an associated method, said input handler, output handler, and signal handler being operative for sending and receiving communication signals to or from another molecule or logic web externally of the respective molecule and being operatively connected to the other micro-
said at least one input handler being operative for queuing input data, said interface handler being operative for determining when a predefined input condition for required input data to be received by said input handler is fulfilled and then invoking said method handler, said method handler being operative for invoking said associated method for processing the input data, and said at least one output handler being operative for outputting a result of the processing of input data by said method.	components, said at least one input handler being operative for queuing input data, said interface handler being operative for determining when a predefined input conditional for required input data to be received by said input handler is fulfilled and then invoking said method handler, said method handler being operative for invoking said associated method for processing the input data, and said at least one output handler being operative for outputting a result of the processing of input data by said method;
	storing the created molecules in a library for

	later run time use;
	deploying on a given computing resource a logic web comprising a plurality of molecules selected to perform a given processing task by invoking a first molecule to be retrieved from the library and executed on the given computing resource; and
	said first molecule invoking one or more other molecules to incrementally extend said logic web "on the fly" by said signal handler of said first molecule sending a communication signal to invoke said other molecule(s).
35. A distributed computing method comprising the steps of: **creating a plurality of software entities ("molecules") each of which is to be deployed in a remote computing environment and is configured with signal handling** software micro-components	1. A computer processing method comprising the steps of: **creating a plurality of software entities ("molecules") each of which is configured with software micro-components including a signal handler,** at least one input handler, at least one output handler, an interface handler, and at least one method handler for an associated method, said input handler, output handler, and signal handler being operative

for sending and receiving communication signals to or from a source externally of the respective molecule said signal handling micro-components of each molecule being operatively connected a selected one of a plurality of method handling software micro-components for processing input data by a selected method in the given remote computing environment in which said molecule is deployed and providing **a resulting output of processing the input data...**	**for sending and receiving communication signals to or from another molecule or logic web externally of the respective molecule and being operatively connected to the other micro-components,** said at least one input handler being operative for queuing input data, said interface handler being operative for determining when a predefined input conditional for required input data to be received by said input handler is fulfilled and then invoking said method handler, said method handler being operative for invoking said associated method for processing the input data, and said at least one output handler being operative for outputting **a result of the** processing **of input data by said method**;

The limitation recited in claims 21-26 are obvious variations of limitations in '697 Claims of 1-4.

The limitation recited in claim 35 is obvious variations of limitations in '697 Claim 1.

Claim Rejections - 35 USC § 102

6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the

basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless --
 (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United
States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the
United States before the invention by the applicant for patent, except that an international application filed under the treaty
defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if
the international application designated the United States and was published under Article 21(2) of such treaty in the English
language.

7. Claims 21-29, 31, 33, and 35-38 are rejected under 35 U.S.C. 102(e) as being anticipated

by US Patent No. 6,038,593 to Huckins, hereinafter called Huckins.

Per claims 21 and 28:

Huckins disclose:

-   A distributed computing method (col. 3, lines 11-12 "distributed service architecture")

    comprising the steps of:

-   creating a plurality of software entities ("molecules") (col. 5, lines 51-55 "application 233

    creates the class object residing either in the application's own process space or in another

    process space on the machine 103" also, fig. 2) each of which is to be deployed in a

    remote computing environment (see Fig. 1 and related discussion) and is configured with

    signal handling software micro-components for sending and receiving communication

    signals to or from a source externally of the respective molecule (col. 3, lines 21-24 "The

    RI interface server is for sending and receiving protocol data units (PDUs) to and from

    another machine through the communication pipe linking the machines"), said signal

    handling micro-components of each molecule being operatively connected a selected one

of a plurality of method handling software micro-components for processing input data

by a respective selected method in the given remote computing environment (col. 4, lines

18-20 "The RI interface server is for sending and receiving protocol data units (PDUs) to

and from another machine through the communication pipe linking the machines") in

which said molecule is deployed and providing a resulting output of processing the input

data (col. 6, lines 55-67 "the resulting changes to the data may be displayed to the user

via the graphical user interface (GUI) 208 associated with the first application 233" also

fig. 2);

- deploying a first molecule with a method handling software component for a specific

selected first data processing method (see Fig. 1 and related discussion) on a respective

one of a plurality of remote computing environments (col. 4, lines 4-6 "a communication

pipe 101 is tapped into by multiple machines 103, 105 and 107 in which a copy of the

same application will be running" also fig. 1), together with a library containing a

plurality of method handling software components for a corresponding plurality of data

processing methods which may be selectively invoked in the remote computing

environment (see Fig. 2 and Fig. 3 and related discussion);

- wherein each first molecule deployed its respective computing environment invokes a

next subsequent molecule with a method handling software component retrieved from the

library for implementing a next selected in response to receipt of a communication signal

from the external source (see Fig. 5, and related discussion), thereby creating a "logic

web" of molecules of successive data processing method in successive layers of

incremental processing steps (col. 3, lines 24-28 "The RI interface is for handling

function calls from the application, and for creating an instance of itself, called a local-

implementation (LI) interface, in a remote machine")to incrementally extend said logic

web "on the fly" in response, at least in part, to communication signals from the external

source (col. 3, lines 31-40 "RI server will create a PDU... includes... object

identification (OID) of the RI interface and a unique function ID (FID) identifying the

particular function that was invoked... PDU is... forwarded to... communication pipe...

PDU... received... RI server... the RI server uses the OID and FID contained in the PDU

to invoke the function through the previously created LI interface. A local

implementation of the function in the remote machine is then performed").

Per claim 35:

Huckins disclose:

- A distributed computing method (col. 3, lines 11-12 "distributed service architecture")
  comprising the steps of:

- creating a plurality of software entities ("molecules") (col. 5, lines 51-55 "application 233
  creates the class object residing either in the application's own process space or in another
  process space on the machine 103" also, fig. 2) each of which is to be deployed in a
  remote computing environment (see Fig. 1 and related discussion) and is configured with
  signal handling software micro-components for sending and receiving communication
  signals to or from a source externally of the respective molecule (col. 3, lines 21-24 "The
  RI interface server is for sending and receiving protocol data units (PDUs) to and from
  another machine through the communication pipe linking the machines"), said signal

handling micro-components of each molecule being operatively connected a selected one

of a plurality of method handling software micro-components for processing input data

by a selected method in the given remote computing environment (col. 4, lines 18-20

"The RI interface server is for sending and receiving protocol data units (PDUs) to and

from another machine through the communication pipe linking the machines") in which

said molecule is deployed and providing a resulting output of processing the input data

(col. 6, lines 55-67 "the resulting changes to the data may be displayed to the user via the

graphical user interface (GUI) 208 associated with the first application 233" also fig. 2);

- deploying a first molecule with a method handling software component for a specific

    selected first data processing method (see Fig. 1 and related discussion) on a respective

    one of a plurality of remote computing environments (col. 4, lines 4-6 "a communication

    pipe 101 is tapped into by multiple machines 103, 105 and 107 in which a copy of the

    same application will be running" also fig. 1), together with a library containing a

    plurality of method handling software components for a corresponding plurality of data

    processing methods which may be selectively invoked in the remote computing

    environment (see Fig. 2 and Fig. 3 and related discussion), wherein the remote computing

    environments are network computing sites distributed on a network (see Fig. 1 and

    related discussion);

- wherein each first molecule deployed its respective computing environment invokes a

    next subsequent molecule with a method handling software component retrieved from the

    library for implementing a next selected in response to receipt of a communication signal

    from the external source (see Fig. 5, and related discussion), thereby creating a "logic

web" of molecules of successive data processing methods in successive layers of incremental processing steps (col. 3, lines 24-28 "The RI interface is for handling function calls from the application, and for creating an instance of itself, called a local-implementation (LI) interface, in a remote machine") to incrementally extend said logic web "on the fly" (col. 3, lines 31-40 "RI server will create a PDU... includes... object identification (OID) of the RI interface and a unique function ID (FID) identifying the particular function that was invoked... PDU is... forwarded to... communication pipe... PDU... received... RI server... the RI server uses the OID and FID contained in the PDU to invoke the function through the previously created LI interface. A local implementation of the function in the remote machine is then performed")

- providing a communication signal from a contol source on the network to each logic web at each network computing site to command , at least in part, its performance of more or selected data processing functions "on the fly" in its respective computing environment autonomously (col. 4, lines 15-18 "the communication pipe supports a multicast protocol wherein a machine does not necessarily have a one to one relationship with another machine" also fig. 1), and return of an output which is desired to be obtained from that network computing site (col. 4, lines 4-6 "a communication pipe 101 is tapped into by multiple machines 103, 105 and 107 in which a copy of the same application will be running" also fig. 1).

Per claim 22:

The rejection of claim 21 is incorporated, and further, Huckins disclose:

- wherein each logic web in each computing environment performs its data processing

    functions in respective computing environment autonomously (col. 4, lines 15-18 "the

    communication pipe supports a multicast protocol wherein a machine does not

    necessarily have a one to one relationship with another machine" also fig. 1), and returns

    an output which is desired to be obtained from that computing environment (col. 4, lines

    4-6 "a communication pipe 101 is tapped into by multiple machines 103, 105 and 107 in

    which a copy of the same application will be running" also fig. 1).

Per claims 23, 36, and 38:

The rejection of claims 21 and 35 is incorporated, respectively, and further, Huckins disclose:

- wherein each logic web returns the output for its respective computing environment to an

    external monitoring entity, and said external monitoring entity combines the outputs from

    the other computing environments to obtain a combined output of distributed computing

    (col. 6, lines 65-67 "the resulting changes to the data may be displayed to the user via the

    graphical user interface (GUI) 208 associated with the first application 233")

Per claims 24, 25, and 37:

The rejection of claims 21 and 36 is incorporated, respectively, and further, Huckins disclose:

- wherein the computing environments are a plurality of computing sites distributed on a

    network (col. 3, lines 12-14 "a multicast environment and enables remote application

    control among multiple machines connected by a communication pipe" also fig. 1), and

    the logic webs return their outputs by sending signals on the network (col. 7, lines 37-46

"message will... received by... machines connected to the pipe 101... control message

will contain the new OID "X" and will instruct server 321 to instantiate an associated

interface object 329 (a local-implementation (LI) interface) and to associate it with the

same OID X")

Per claims 26 and 33:

The rejection of claim 21 is incorporated, and further, Huckins disclose:

- wherein said software micro-components include a signal handler, at least one input

  handler, at least one output handler, an interface handler, and at least one method handler

  for an associated method (col. 6, lines 55-67 and col. 7, lines 1-16 also fig. 2),

- said at least one input handler being operative for queuing input data (fig. 2, element

  203), said interface handler being operative for determining when a predefined input

  condition for required input data to be received by said input handler is fulfilled (col. 6,

  lines 18-24 "Application 233 obtains an interface 209 from COM interface server 203,

  having a globally unique identifier GUID(j)... an object identification (OID) X. Server

  203 has a GUID(j)... access to local implementation of the function f(y) with results of

  the performed function being displayed to the user through the graphical user interface

  (GUI) 208") and

- the invoking said method handler, said method handler being operative for invoking said

  associated method for processing the input data (col. 5, lines 47-54 "function identifier

  (FID) that is assigned for each function listed in interface 209... functions are redefined

  by interface definition. Each unique interface, such as IDocument, is identified by a

GUID which the client application 233 and server 203 are aware of. The unique interface

must also be identically defined by both client and server 203"), and

-   said at least one output handler being operative for outputting a result of the processing of

    input data by said method (col. 6, lines 65-67 "the resulting changes to the data may be

    displayed to the user via the graphical user interface (GUI) 208 associated with the first

    application 233")

Per claim 27:

The rejection of claim 21 is incorporated, and further, Huckins disclose:

-   wherein said software micro-components are stored in a library for run time use (col. 8,

    lines 18-20 "the object can reside in application 135 and will cause changes to data stored

    in machine 105"), and during run time a logic web is deployed in a given computing

    environment invoking a first molecule to be retrieved from the library and executed in the

    given computing environment (col. 7, lines 13-17 "GUID for server 301 may be obtained

    by either embedding the GUID value in the binary code of application 133, or by the

    application 133 at run time from its application configuration file or the system registry

    of the operating system"), and said first molecule invoking one or more other molecules

    to incrementally extend said logic web "on the fly" (col. 3, lines 31-40 "RI server will

    create a PDU... includes... object identification (OID) of the RI interface and a unique

    function ID (FID) identifying the particular function that was invoked... PDU is...

    forwarded to... communication pipe... PDU... received... RI server... the RI server uses

    the OID and FID contained in the PDU to invoke the function through the previously

created LI interface. A local implementation of the function in the remote machine is then performed")

Per claim 29:

The rejection of claim 21 is incorporated, and further, Huckins disclose:

- wherein said creating step includes creating molecules having a built-in handler function for performing a clean-up of its functions when the molecule is to be terminated (col. 7, lines 21-27 "the switcher task would simply modify the system registry or application configuration file for application 133 to specify GUID(i) of the "distributed" server 301 as the server GUID that application 133 should use for its function calls. To switch back to local mode, the switcher task would reset the configuration file to GUID(j) of the "non-distributed" server 303")

Per claim 31:

The rejection of claim 21 is incorporated, and further, Huckins disclose:

- wherein said signal handler can receive signals for and has a handler type for dynamically reconfiguring the micro-component handlers of the molecule while in existence to perform a processing task (col. 3, lines 36-40 "PDU is received by a RI server in a remote machine, the RI server uses the OID and FID contained in the PDU to invoke the function through the previously created LI interface. A local implementation of the function in the remote machine is then performed")

Substantially as claimed.

Claim Rejections - 35 USC § 103

1.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2.   Claims 30, 32, 34, 39 and 40 are rejected under 35 U.S.C. 103(a) as being unpatentable

over Huckins in view of US Patent No. 5,701,439 to James et al., hereinafter called

James.

Per claim 30:

The rejection of claim 21 is incorporated, and further, Huckins disclose wherein said creating

step includes creating molecules having a handler type for recording information on the state of

its micro-component handlers and signaling such state information externally through said signal

handler.

However, James discloses in an analogous computer system a transition diagram showing

various states that the system passes through during a normal simulation run (col. 4, lines 58-63

"While in operate..., be desirable to temporarily suspend execution of the simulation, then

subsequently return to the operate mode... at some point in time the operate mode will normally

be terminated in some orderly fashion, possibly to return to an initialization state. This

sequence of states and state transitions is detailed in FIG. 2" also fig. 2).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time

the invention was made to incorporate the method of recording different stats of the program as

taught by James into the method of distributing applications for multicast environment systems

as taught by Huckins. The modification would be obvious because of one of ordinary skill in the

art would be motivated to use stat information to provide data handling in simulation as

suggested by James (col. 1, lines 62-67 to col. 2, lines 1-8).

Per claims 32 and 39:

The rejection of claims 21 and 38 is incorporated, respectively, and further, Huckins does not

explicitly disclose wherein said interface handler includes a handler type for providing the

molecule with the characteristic of autonomously waiting, looking, and proceeding with said

associated method for processing input data by waiting until said input handler indicates that the

predefined input conditions are present before invoking said method handler for the associated

method.

However, James discloses in an analogous computer system a wait queue being used by

discrete-event components of the system (col. 4, lines 36-40 "discrete-event components of the

system use a set of execution (event) 22, wait 28, and resource 26 queues for each discrete-

event process; a single continuous model execution (consim) 24 queue is shared by all

continuous model processes" also fig.1, element 28).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time

the invention was made to incorporate the method of using wait queue feature as taught by

James into the method of distributing applications for multicast environment systems as taught

by Huckins. The modification would be obvious because of one of ordinary skill in the art

would be motivated to implement a wait queue in the system to improve handling of functions

calls through the interface for uniform simulation as suggested by James (col. 1, lines 62-67 to

col. 2, lines 1-8).

Per claims 34 and 40:

The rejection of claims 21 and 39 is incorporated, respectively, and further, Huckins does not

explicitly disclose wherein said input handler is selected from one of a plurality of input handler

types corresponding respectively to a plurality of different data source types.

However, James discloses in an analogous computer system the architecture of the

combined discrete event and continuous model simulation permits the use of several language

models with separate process for each function (col. 4, lines 34-36 "global data and control

structures, designated as 20 in FIG. 1, reside in shared memory accessible to any process that

requires them" also fig. 1).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time

the invention was made to incorporate the method of using multiple language feature as taught

by James into the method of distributing applications for multicast environment systems as

taught by Huckins. The modification would be obvious because of one of ordinary skill in the

art would be motivated to implement a multiple language feature in the system to improve

handling of functions calls through the interface for uniform simulation as suggested by James

(col. 1, lines 62-67 to col. 2, lines 1-8).

Response to Arguments

8. Applicant's arguments with respect to claim 21 and 35 have been considered but they are

    not persuasive.

In the remarks, the applicant has argued that:

-   The invention as claimed enables the local "logic web" to implement any selected data

    handling method out of a plurality of data handling methods stored in a library on the

    local machine, so that the logic web incrementally extended "on the fly" in response to

    communication signals sent from an external source. This is a distinctly different method

    of "on the fly" remote computing not disclosed or suggested in the Huckins patent.

Examiner's response:

1. Regarding the limitations logic web to implement any selected data handling method out

    of a plurality of data handling methods stored in a library and logic web incrementally

    extended "on the fly", Huckins discloses distributed service architecture that enables

    remote application control among the multiple machine connected by a communication

    pipe. The Remote implementation (RI) interface, which provides the function, called by

    the application and seding and receiving data units to and from another machine. In doing

    so, the system does provide a logic web and on the fly communication (see the rejection

    above and Fig. 1 and 2 and related discussion). Applicant only makes general allegations

    and does not point out any errors in the rejection. Therefore, the rejection is proper and

    maintained herein.

Conclusion

1.   The prior art made of record and not relied upon is considered pertinent to applicant's

disclosure.

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to **Satish S. Rampuria** whose telephone number is **(571) 272-3732**.

The examiner can normally be reached on **8:30 am to 5:00 pm** Monday to Friday except every

other Friday and federal holidays. Any inquiry of a general nature or relating to the status of this

application should be directed to the **TC 2100 Group receptionist: 571-272-2100**

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, **Tuan Q. Dam** can be reached on **(571) 272-3695**. The fax phone number for the

organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Satish S. Rampuria
Patent Examiner
Art Unit **2191**
05/02/2005

ANIL KHATRI
PRIMARY EXAMINER